Knowledge Sharing East & West:
All Requirements Are Not Equal   by Todd Post

Todd Post is the editor of ASK Magazine and works for Edutech Ltd out of Silver Spring, Maryland. He has written for many publications in print and online.

In my last Conference Report, I told you about the KS East and West Meetings in December. March 12-15 found KS Participants back in Atlanta for the East meeting and, instead of San Francisco, we were in Pasadena this time for the West.

Requirements figured heavily on the agenda in both locations. At East and West, Walker Royce, Vice President and General Manager at Rational Software Corporation and author of *Software Project Management*, delivered a lively keynote on the management renaissance taking place within the software industry. 'Paradigm shift' is generally a facile expression to describe any kind of change taking place these days, but if Royce is reconnoitering the software industry with 20-20 vision that usage here might be right on target.

What Royce sees is an industry, hardly by itself, doing things faster than ever

Author and project manager Walker Royce gave presentations in both Atlanta and Pasadena for participants at Knowledge Sharing East and West.

before, with the traditional approach to the software development lifecycle process just no longer able to keep up. Traditionally, requirements are defined early in the development process and done with after that. Royce says this was never really satisfactory to anyone, but then that's another matter.

What he has observed taking place instead of the traditional approach is an iterative model, whereby requirements are redefined simultaneously during design and development. In his book, Royce lays out the foundation of this new approach by categorizing it into ten principles, the principles of modern software management, and these are:

1. Focus the process on the architecture first
2. Attack risks early with an iterative life cycle
3. Emphasize component-based development
4. Change management of all artifacts

5. Simplify change freedom with round-trip engineering
6. Use rigorous, model-based design notation
7. Instrument the process for objective quality control
8. Emphasize demonstration-based assessment
9. Plan releases with evolving levels of detail
10. Establish a scalable, configurable process

APPL **ASK**

Naturally, we heard a lot about all ten principles at the meetings. Royce is a commanding speaker, and he brought impressive evidence to back up his arguments that a change is upon us, and that when applied right will make a difference for managers who are keen on implementing it.

How applicable is all this to NASA project managers? Many who attended seemed to think it was very, as they peppered Royce with questions about how to apply an iterative approach to their own scenarios.

Can it work for you? It's all in the application. Remember principle 10: *Establish a scalable, configurable process*. In his book, Royce called this 'Tailoring the process!' That is, it all depends on the situation. Context is the key.

Alex Laufer picked up on this theme in his presentation, "Tailoring Project Processes," delivered in both Atlanta and Pasadena, which included several examples of how successful project managers from the Navy and Coast Guard, as well as other federal agencies, practice tailoring. The interesting questions raised by

ASK Editor-in-Chief Alex Laufer and Managing Editor Michelle Collins strategizing.

Laufer's presentation went far beyond tailoring any one situation to rethinking what organizations might be like if tailoring were accepted as the appropriate response to the amazing variety of situations managers confront daily.

In Atlanta we also heard from Linda Rosenberg, one of NASA's own software experts, who presented an exercise on defining requirements. (See Michelle Collins' story in this issue, *Lessons From the Great Masters*) The problems with most requirements are mirrored in their documentation, she argued. At one point Rosenberg displayed a graphic depicting a bizarre amalgam of a zebra, a cat, and a skunk. "If you don't do requirements properly, this is what you get," she said, an unrecognizable species of beast you're not sure whether to offer up to science or straight out euthanize.

Then in Pasadena we heard another take on requirements from John Allmen, Deputy Division Chief of the Space Projects Division at NASA's Ames Research Center. Allmen's presentation "Better: the Enemy of Good Enough?" focused

heavily on how to stave off requirements creep. (See how Allmen converted his presentation into a Practice for this issue of ASK on the following pages) Like Royce and Rosenberg, Allmen believes clear requirements play a critical role in a project's success. Allmen would allow mangers room to improve on requirements, but he would also treat each improvement as a micro project with a separate budget and schedule of its own. First meet the minimum, then you can press on. "A requirement is intended to tell you when you've finished your job on a particular aspect of the project: Good Enough. Go for a better requirement if you know you can afford to, but remember to maintain sight of the original requirement and its original budget allocation as you make better."

Overall there was plenty to learn from the four speakers, and from the other NASA project managers who spoke at the conference and whose stories will probably be featured in one of the future issues of ASK

Talk with you again after the Master's Forum over the summer.

Knowledge Sharing East presenter Linda Rosenberg and Jo Gunderson enjoy a break between presentations in Atlanta.

In praise of Walter Ryce's book **Software Project Management,** Barry Boehm, Director of the Center for Software Engineering at the University of Southern California, writes "It has been difficult to find a fully articulated alternative management approach for dealing with such issues as commercial component integration, software reuse, and risk management. This book provides a new experience-tested framework and set of guidelines on how to proceed."